Completa el Manual Usuario Testing (deberás replicar la tabla x las veces que compruebes una funcionaldidad):

|  |  |  |  |
| --- | --- | --- | --- |
| Nombre Caso de Prueba | Iniciar sesión y visualizar ítems | | |
| Nombre de Usuario Testing | Connie Palacios | |  |
| Correo | demo2@demo.cl | |  |
| Teléfono | n/a |  |  |
| Epica 1 | Fecha 6-10-25 | Hora 17:45 |  |
| HU1 | Como cliente, quiero iniciar sesión para usar el sistema | | |
| HU2 | Como cliente, quiero visualizar el listado de ítems con su precio y estado | | |
| HUn | Mantener sesión activa tras recargar la página | | |
| Tipo de Usuario | Cliente (ROL=CLIENT/ADMIN; se prueba CLIENTE) | | |
| Artefacto a probar | Front web (Angular: páginas /login y /items) + API (Node/Express): POST /api/auth/login, GET /api/items | | |
| Prueba a realizar | Prueba funcional + experiencia de usuario del flujo básico (login → listado de ítems) | | |
| Tiempo de duración | 10–12 minutos | | |
| Actividad a Realizar | Info. del paso a ejecutar | | Registro de errores/Corrección |
| Paso 1 | Abrir http://localhost:4200/login. Debe cargar el formulario “Iniciar sesión”. | | Si no carga, verificar que Angular esté corriendo (ng serve) y que no existan errores en consola del navegador |
| Paso 2 | Ingresar email demo2@demo.cl y contraseña Demo\*123. Hacer clic en Ingresar. | | Si credenciales inválidas: la API responde {"ok":false,"error":"Credenciales"}. Corregir usuario/clave o recrear usuario vía /admin o API. |
| Paso 3 | Tras login exitoso, verificar que se guarda el token JWT (localStorage) y la app redirige a /items. | | Si no redirige, revisar guard/route en Angular y respuesta 200 del endpoint POST /api/auth/login |
| Paso 4 | |  | | --- | |  |  |  | | --- | | En /items, el front consume GET /api/items. Debe mostrar >0 registros con NAME, DESCRIPTION, PRICE, ACTIVE. | | | Si aparece “No hay datos para mostrar”: comprobar API (Invoke-RestMethod /api/items), DB Oracle desbloqueada (ORA-28000 → ALTER USER HOTEL\_APP ACCOUNT UNLOCK;). |
| Paso 5 | Validar que los datos coinciden con Oracle (ej.: “Hab. Simple – 45”). | | Si hay diferencias, correr SELECT id,name,price,active FROM items ORDER BY id; y revisar mappeo en items.js (backend) y render en Angular. |
| Paso 6 | Presionar F5 (recarga). Debe **mantener sesión** y permanecer en /items. | | Si expulsa a /login, revisar persistencia y verificación del token en el guard del front |
| Paso 7 | Cerrar sesión (si tienes botón). Debe borrar token y llevar a /login | | Si no vuelve a login, limpiar manualmente el storage y verificar handler del botón logout. |
| Observaciones | | |  |
| * En la creación de usuarios, si aparece EMAIL\_DUPLICADO, validar constraint UNIQUE en USERS(EMAIL) y usar otro correo. * Si el backend muestra ORA-1017/ORA-28000, revisar .env del backend (usuario/clave Oracle) o desbloquear cuenta. * En desarrollo se usa proxy de Angular para /api a http://localhost:4000 (evitar CORS). | | | |